quantum simulation, Poisson Pareto Burst Process

Quantum Simulation - Rare Event Simulation by means of Cloning, Thinning and Distortion

R. G. ADDIE
University of Southern Queensland
Toowoomba, Australia
addie@usq.edu.au

January 24, 2003

Abstract. A method of rare event simulation, termed here *quantum simulation*, and known also (with some variations) as *population Monte Carlo*, and *Sequential Markov Chain simulation*, is applied in this paper to *rare event simulation* of communication systems.

The technique described in this paper generalizes *importance sampling*, *importance splitting* and the Monte Carlo method which uses a collection (population) of particles.

The term *quantum simulation* is used for the rare-event simulation technique presented in this paper because the entire ensemble of simulations resembles the parallel universes model of quantum mechanics. By using cloning, thinning and distortion it is possible to design simulations with the speed of importance sampling and the flexibility of importance splitting.

A particularly difficult system is investigated in this paper by means of quantum simulation, namely a buffer fed by a Poisson-Pareto Burst Process. The simulations are able to confirm an analytic approximation for this model to a degree not previously achievable. Furthermore, this approximation was originally developed as a consequence of the investigation of this model by means of quantum simulation.

Keywords: Poisson-Pareto Burst Process, Long-range dependence, Importance Splitting, Importance Sampling, Quantum Simulation

1 INTRODUCTION

Quantum simulation, population Monte-Carlo, sequential Markov chain simulation, ...[13, 7, 11, 6, 1] makes use of spontaneous generation of clones (copies) of simulation processes which then proceed with an independent random number stream. Processes are also thinned (killed) to ensure that the total number of processes stays within reasonable bounds or remains constant. The cloning rate may be state dependent, and in particular cloning rates may be chosen in such a way that events of interest occur more frequently.

Individual threads (processes, clones) do not necessarily progress with the same dynamics as the original system being modelled although as an aggregate, the entire collection of threads can always be viewed, by using an appropriate transformation of the statistics of the collection of threads, as an unbiased model of the original system.

This method of simulation is a generalisation of *importance splitting*, also known as the *Restart Method*, as introduced in [18] and further developed in, for example, [5, 10, 12]. Quantum simulation is also a generalisation of *importance sampling*, eg [8, 14].

Quantum simulation provides a framework in which both importance splitting and importance sampling can be described. Importance sampling makes use of an analytic formula for a *change of measure* which transforms ("distorts") the model under consideration into one which can be simulated more quickly.

In the case of quantum simulation, an arbitrary change of measure can also be introduced, in addition to cloning and thinning, and a formula for the conversion of the statistics collected from the simulation back into those appropriate for the original model, will always be available, although it is not necessarily a Radon-Nikodym derivative, as it would be in the case of importance sampling. In the case of quantum simulation the inverse transformation formula is calculated by the simulation program.

The papers [5, 12] on a method known as *Direct Probability Redistribution*, a method which computes a suc-

cession of simulations according to the RESTART technique but computes statistics by means of accumulation of weights rather than by estimation of conditional distributions.

Each *thread*, the *i*th, say, of a quantum simulation is accompanied by a number, its weight, $p_i(t)$ (the weight of thread i at time t). The general rule which applies to unbiased estimation in a quantum simulation can be expressed as follows. Suppose that the original process is X_t and the quantum simulation contains k threads, $X_t^{[1]}, \ldots, X_t^{[k]}$, with weights $p_1(t), \ldots, p_k(t)$ at time t. Suppose that f is an arbitrary function on the state space of the process X_t. Then

$$E\left(f(X_t)\right) = \sum_{i=1}^{k} E\left(p_i(t) f(X_t^{[i]})\right), \tag{1}$$

for all $t > 0$. In fact, we shall require that this equation holds even if the function f depends on previous values of the process X_t up to time t,so long as it is a measurable function on the path of the process. A more precise statement of this rule will be given below, in equation (3). This equation and its use as the defining property of quantum simulation is the most significant contribution of this paper.

A quantum simulation which has exactly one thread at any time is precisely equivalent to importance sampling because (3) becomes equivalent in this case to the defining rule of an importance sampling simulation, which specifies that the correction factor must be a certain Radon-Nikodym derivative.

The most general possible classes of distortion, cloning and thinning are not readily stated. Furthermore, although at first it may seem that a particular extension of a technique of cloning, or thinning, or distortion might seem to be relatively minor and unimportant, there are some applications which are virtually intractable unless the richest possible collection of techniques is available. The main problem considered in this paper is an example of such a problem. The hypothesis proposed and supported by this paper is that the best way to define the full range of valid techniques is to allow any simulation, in the form of a collection of threads, which satisfies the consistency principle (3) which has already been cited.

More than one example is considered in the sequel, although all examples come from the field of communication networks. The main example considered is one which has proved to be very difficult to analyse by conventional simulation or by means of mathematical analysis. It is the buffering (at a server in a communication network, e.g. a link or a router) of traffic which forms a Poisson Pareto Burst Process (PPBP). The PPBP is acknowledged by many authors to be a good model of many types of network traffic. Several authors have tackled the analysis of buffering in network elements which have to carry this traffic [16, 15, 9, 17], although the best of the available results, until recently, appear to provide upper and lower bounds which are rather widely separated in situations of the greatest interest. More recently, in parallel with the present work, a more accurate approximation of the buffer exceedence probabilities of a Poisson-Pareto queueing system has been achieved in [4].

2 DEFINITION OF QUANTUM SIMULATION

Although the primary interest in this paper is in simulations, it will be convenient to view each simulation as a stochastic process, and likewise a quantum simulation as a new type of stochastic process, a *quantum stochastic process*, or, for short, a QSP.

Definition 2.1 *A* quantum stochastic process *is a collection of stochastic processes, $\{X_t^{(i)}\}_{t\in[s_i,f_i]}$, $i \in I$, taking values in a state space Σ, with measurable sets $\mathcal{S}$, together with their* weights, *$p_i(t)$, $i \in I$, and a* prior function, *$\phi : I \to I$ (which indicates, for each process, which process* precedes *it).*

These weights always have the property

$$\sum_{i\in I \& s_i \le t < f_i} E\left(p_i(t)\right) = 1, \qquad t \ge 0. \tag{2}$$

It will also always be required that a quantum simulation remain consistent with goal of making unbiased estimates of a real system (or a real simulation). In Subsection 2.1, we shall set out the conditions for this *consistency* to hold (in equation (3)) and (2) shall be seen as a consequence of this consistency condition.

A quantum stochastic process such as the one just defined will be denoted by $\left\{\{X_t^{(i)}\}, p_i(\cdot), \phi : t \in [s_i, f_i],\ i \in I\right\}$.

The index set, I, is always finite and at any time, t, we expect the total number of *active* simulations to be significantly less than the total number of elements in I.

When one stochastic process (or simulation) stops, in many cases, one or more other stochastic processes will continue from where this one left off. For this reason we need a mapping, $\phi : I \to I$, which designates, for each *thread* (as we shall call the stochastic processes, in order to indicate that each is merely a component of a larger view), of which *prior* thread this thread is a continuation. Thus, for any $i \in I$, $j = \phi(i)$ is another thread such that $f_j = s_i$. There may be more than one thread, i, such that $j = \phi(i)$, which is meant to indicate that the thread j has *cloned* a collection of *children* (or clones). The sum of the weights of the collection of all the children of any thread which has children should equal the weight of the parent thread, i.e.

$$\sum\{p_i(s_i+) : \phi(i) = j\} = p_j(f_j-),$$

where $t-$ represents a time just before t.

It is also possible that a thread may terminate and not leave behind any children, in which case, the weight of the terminating thread will need to be distributed amongst some other threads.

2.1 CONSISTENCY PROPERTY

We want a quantum stochastic process to be able to substitute for a normal stochastic process, i.e. any use to which a conventional process (or simulation) can be put, there should be a standard way to use a quantum stochastic process in the same way. The *consistency property* defined below, in Definition 2.3 is the condition which ensures that this is the case..

Definition 2.2 *Let $I(t)$ denote $\{i \in I : s_i \le t < f_i\}$, and for each $i \in I(t)$ define the stochastic process $\{X_t^{\{i\}}\}$ as the concatenation of the thread i together with the sequence of successive* prior *threads.*

The space of measurable functions from $[0,T]$ to Σ is denoted by $\Sigma^{[0,T]}$. Thus, for any element, $f \in \Sigma^{[0,T]}$, $f : [0,T] \to \Sigma$ and for all $U \in \mathcal{S}$, $f^{-1}(U) \in \mathcal{B}(\mathbb{R})$, the Borel sets of $\mathbb{R}$. The measurable sets in $\Sigma^{[0,T]}$ are logically defined as the collection $\mathcal{S}^{\mathcal{B}([0,T])}$ of sets generated, under countable unions and intersections, from the sets of the form

$$\{f : f(t_1) \in A\}$$

as t_1 varies over $[0,T]$ and A varies over $\mathcal{S}$.

Any Σ-valued stochastic process defined on a probability space $(\Omega, \mathcal{F}, P)$, and $\{X_t\}$ in particular, can be viewed as a measurable mapping from Ω to $[0,T]^{\Sigma}$. For clarity, when a stochastic process such as $\{X_t\}$ is viewed in this way, we shall denote it simply by X. Similarly, the stochastic process $\{X_t^{\{i\}}\}$ when viewed in this way will be denoted by $X^{\{i\}}$.

Definition 2.3 *A quantum stochastic process is* consistent *with the stochastic process $\{X_t\}$ if, for all $t \in \mathbb{R}$, $F \in \mathcal{S}^{\mathcal{B}([0,t])}$,*

$$E\left(\sum_{i \in I_t} \left\{p_i(t) : \omega \in (X^{\{i\}})^{-1}(F)\right\}\right) = P\left\{X^{-1}(F)\right\}. \quad (3)$$

For example, if we choose the universal set, $\Sigma^{[0,T]}$, for F, (3) implies

$$E\left(\sum_{i \in I_t} p_i(t)\right) = 1,$$

as we flagged earlier would be required in a QSP.

An equivalent statement of this condition is as follows. Suppose Z is an arbitrary measurable mapping from $\Sigma^{[0,T]}$ to $\mathbb{R}$. Then, we require

$$E\left(\sum_{i \in I_t} \left\{p_i(t) \times Z(X^{\{i\}})\right\}\right) = E\left\{Z(X)\right\}. \quad (4)$$

That is to say, expectations of random variables defined on the QSP (which we take to be done as on the LHS of (4)) should be consistent with expectations of random variables defined on the original process.

2.2 A CALCULUS OF QUANTUM STOCHASTIC PROCESSES

Quantum stochastic processes can be added together, multiplied, in fact any functional combination of QSPs can be formed, they can be *merged*, and modified by splitting, thinning, and distortion. In this sense, they form a generalization of the concept of a conventional stochastic process which is complete in the sense that all the operations one might apply to a stochastic process or collection of stochastic processes can also be applied to QSPs. For more details of the calculus of quantum stochastic processes, see [2].

One particular method of modification of a QSP warrants attention here:

2.2.1 Distortion with Preserved Total Weight

This technique is a rendition of importance sampling which includes a mechanism for *preserving total weight* of the collection of threads.

Consider a QSP $\left\{\{X_t^{(i)}\}, p_i(\cdot), \phi : t \in [s_i, f_i],\ i \in I\right\}$, where $I = \{1, \ldots, I\}$, with state space Σ, which is consistent with the conventional stochastic process $\{X_t\}$ which has the stationary probability measure π on $\Sigma^{[0,T]}$. More specifically, let us suppose that $X_t^{[1]} \sim \pi$ is identically distributed to $\{X_t\}$ while the remaining threads, $X_t^{[i]}$, $i = 2, \ldots, I$ are identically distributed to the process $\{Y_t\}$ which has probability measure γ on $\Sigma^{[0,T]}$. The natural projection of π onto $\Sigma^{[0,t]}$ will be denoted by π_t, and similarly for γ, γ_t. That is to say, π_t is the probability measure of a stochastic process on the interval $[0,t]$ which is identical to the process $\{X_t\}$ on this sub-interval of $[0,T]$.

Furthermore, suppose that the Radon-Nikodym derivative,

$$\psi_t = \frac{d\pi_t}{d\gamma_t} > 0, \qquad j = 1, \ldots, J,$$

is non-zero (and positive) for all $t > 0$.

To complete the definition of this QSP, we now set

$$p_i(t) = \begin{cases} \frac{1}{I}\psi_t(\{X_\tau^{\{i\}}\}_{\tau=0}^t), & i > 1 \\ 1 - \frac{1}{I}\sum_{i=2}^I p_i, & i = 1. \end{cases}$$

Notice the use of the process $\{X_\tau^{\{i\}}\}$. The prior function, ϕ has implicitly been used here in the definition of this stochastic process. But it is the setting for p_1 which is of most interest here. The other values are as we might naturally expect in an importance sampling simulation. The Radon-Nikodym derivatives correct for the distortion of the law of evolution of the processes $X_t^{(i)}$, $i = 2, \ldots, I$ and we have an additional factor of $\frac{1}{I}$ simply because we have I simultaneous threads rather than just one.

Since $E\{\frac{d\pi_t}{d\gamma_t}\} = 1$ for all t,

$$E\left\{1 - \sum_{i=2}^I p_i\right\} = 0,$$

for all t also and it follows from this (perhaps not so obviously) that (3) is satisfied. However the consequences of including the non-distorted thread are not trivial. In most cases of interest, the weights $p_i(t)$ will tend to zero. This is only natural considering that the distortion has the effect of increasing the frequency of occurrence of certain unlikely events. However, after a certain point, the beneficial aspect of the distortion becomes swamped by the loss of weight of these threads – the simulation no longer generates any observations which usefully contribute to estimates of interest. The undistorted thread however, is able to sop up the weight which is being lost by the distorted thread, and so the simulation continues to generate useful observations, although only at the efficiency level of a conventional simulation.

2.2.2 Distortion with Cloning and Thinning

The problem with the QSP presented in the last subsection is that although it is not losing weight, after a relatively short period, its statistical efficiency reverts to that of a conventional simulation. However, by introducing cloning and thinning to this simulation we can ensure that as many as possible of our threads are hovering in the *sweet spot* where they generate useful statistics at the best possible rate.

Continuing with the QSP

$$\left\{\{X_t^{(i)}\}, p_i(\cdot), \phi : t \in [s_i, f_i],\ i \in I\right\},$$

let us now suppose that when the total weight of the threads 2 to I falls below a certain level we terminate one thread, chosen randomly, and replace this thread by a clone of the main simulation. In this way, the QSP can reach a stationary state where it generates statistics at maximum efficiency for an indefinite period of time. This is the method which is used in the simulations in Section 5 below.

3 CONSISTENCY RULES FOR CLONING AND THINNING

3.1 RULES FOR CLONING AND THINNING WHICH ENSURE CONSISTENCY

Theorem 3.1 *Suppose a quantum stochastic process* $\left\{\{X_t^{[i]}\}_t : i \in I\right\}$ *is consistent with a certain stochastic process* $\{X_t\}$. *We now modify this quantum stochastic process by cloning and thinning, to produce a quantum stochastic process* $\left\{\{\widetilde{X}_t^{[i]}\}_t : i \in \widetilde{I}\right\}$, *with weights* $\widetilde{p}_i(t)$, $i \in \widetilde{I}$.

Suppose that the the index set of the process $\widetilde{X}$ *is* $\widetilde{I} = I \cup \mathcal{J}$, *and the weights* $\widetilde{p}_i(t)$, $i \in \widetilde{I}$, *are the same as the original weights up till the time when cloning or thinning occurs and that when cloning and thinning occurs these weights are changed in accordance with the following constraints:*

(i) after cloning the weight of a cloned thread is divided arbitrarily amongst its clones;

(ii) if any thinning occurs, there are at least 2 *possible* candidates *for thinning;*

(iii) the procedure for selecting whether a thread is a candidate is exactly the same for each thread (it may depend on the history of this thread, or its weight, and possibly depends upon the history of the other threads, in a manner which is symmetric with respect to the other threads);

(iv) the weights of threads which were not candidates for thinning remain the same after thinning;

(v) the sum *of the weights of the candidates which are not thinned, after these weights are adjusted, equals the sum of the weights of the candidates before the thinning;*

(vi) the subset of the set of candidate processes which are actually thinned is selected *randomly* in such a way that the quantum stochastic estimators from the remaining threads in the candidate set are unbiased estimators of the corresponding quantities in the full candidate set*;*

Under these conditions, the quantum stochastic process $\left\{\{\widetilde{X}_t^{[i]}\}_t : i \in \widetilde{I}\right\}$ *is also consistent with* $\{X_t\}$.

Item (vi) is the important condition here; the other conditions are reasonably natural. However, it is appropriate to explain how the requirement (vi) will normally be satisfied. Suppose the candidate threads are $t_1, \ldots, t_k$, and they have weights $p_1, \ldots, p_k$. Now suppose that we intend to select just one thread to remain, after thinning. According to this particular rule, we *must* select the thread which is *not* thinned randomly with probability proportional to its weight. No other procedure for selecting the thread which remains would satisfy condition (vi) (assuming that the other conditions also hold).

On the other hand, suppose that we want to select precisely one thread to be *thinned*. We must do this by selecting each thread in proportion to the sum of the *other* threads in the set of candidates. Again, no other procedure for thinning a single thread would satisfy Rule (vi).

If we wish to select several threads for thinning, we could randomly select the number of threads to be thinned and then use the procedure for thinning one thread repeatedly, or we could select the threads to remain one by one, sampling from the set of candidates without replacement and with probabilities proportional to the weight of each thread.

There are two reasonably natural, different ways to thin the candidate set: random selection of the threads to remain, and random selection of the threads which are to be thinned, using appropriate weights in each case. But

these are not the only algorithms which respect Rule (vi) and there are reasons for wanting to use more complicated procedures for thinning although none that we have reason to explore in this paper.

If the QSP is to remain consistent with the original process, Rule (vi) or something quite similar must be adopted.

Proof

Without loss of generality we can assume that cloning, if any, occurs momentarily before thinning. The rule for weights after cloning is obviously sufficient to ensure consistency of the quantum simulation estimate because the estimates obtained from two cloned threads are identical to each other. This rule could be violated in many cases without causing a lack of consistency – for example, if the existing threads were all statistically identical, varying the total weight assigned to a collection of clones would not upset consistency of the estimator, so long as the weight lost or gained was redistributed.

Now consider thinning, and, in particular, the estimates of $E(Z)$, where Z is as random variable defined in terms of the path(s) of the process(es) $\{X_t\}$ (and therefore also in terms of $\{X_t^{\{i\}}\}$). Suppose the candidates for thinning form the set C at time $\tau-$, and the other processes which are active at that time form the set D. Then, at this time,

$$E\left(\sum_{i\in C\cup D} p_i(\tau-)Z(i)\right) = E\left(\sum_{i\in C} p_i(\tau-)Z(i)\right) + E\left(\sum_{i\in D} p_i(\tau-)Z(i)\right), \quad (5)$$

in which $Z(i)$ denotes the value of Z on thread i.

Since, when thinning occurs, the weights of the processes which are not candidates do not change, the second part of this sum is unchanged as we transit to the point $\tau+$ in time. Let us denote theset of processes which are terminated by C_0. Recall that this set of threads is chosen *randomly*, from the set C, and in a manner such that the quantum stochastic estimate based on the randomly selected subset is consistent with the original quantum stochastic estimate based on the full set C.

Hence, the first sum in (5) *after* the thinning (so, at time $\tau+$) evaluates to

$$E\left(\sum_{t\in C-C_0} p_t(\tau+)Z(t)\right) = E\left(\sum_{t\in C} p_t(\tau-)Z(t)\right).$$

where we have used property (vi). It follows that the quantum simulation *after* the thinning is still consistent with $\{X_t\}$, which concludes the proof. □

4 RELATIONSHIPS BETWEEN DIFFERENT APPROACHES TO RARE EVENT SIMULATION

4.1 QUANTUM SIMULATION GENERALISES IMPORTANCE SPLITTING

In a RESTART simulation, the state space is usually divided into a number of *level sets*, Λ_0, Λ_1, ... In the notation of [12], simulations which enter Λ_0, are restarted at the point of entry n_0 times, those which enter Λ_1 are restarted n_1 times, and so on, and statistical estimates in these simulations are obtained by dividing each contribution to an estimate by the *oversampling factor*, $n_0 \times n_1 \ldots$.

In a QS which is set up exactly along the lines of a RESTART simulation, so that cloning occurs on first entry to Λ_0, and exactly n_0 clones are made at this time, and so on, this oversampling factor will be exactly the inverse of the weight, $p_i(t)$, and hence the estimates obtained from a quantum simulation and the RESTART method will be identical.

The approach to killing un-interesting threads in the RESTART method is likely to be a little different than that recommended above for quantum simulation, which could lead to numerical differences between quantum simulations and the RESTART method, although these differences may be small if terminations only occur when samples have very low weight. Also, if the simulated process is Markov with a finite state space it should be possible to formulate conditions for terminating samples which avoid bias. However, in the non-Markovian case it appears that the best method to avoid bias may be formulate terminations as a randomized process and redistribute weight of the terminated threads in the manner discussed in Theorem 3.1.

4.2 IMPORTANCE SAMPLING AS A GENERALIZATION OF IMPORTANCE SPLITTING

The suggestion is sometimes made that importance splitting is a special type of importance sampling. A justification of this point of view is provided in [12]. However, there are limits to the validity of the this particular way of interpreting importance splitting style simulations as importance sampling simulations which mean that it would not at all be appropriate to adopt the view that in practical terms importance splitting is a special case of importance sampling.

In fact, the contrary view can also be put, that importance sampling is a special case of importance splitting. It can be shown (see [2]) that any importance sampling model can be approximated to arbitrarily good accuracy by an importance splitting type of simulation. This reduction of importance sampling to importance splitting can be achieved by multiple sampling in a controlled manner: at each micro-step of the simulation, we can form many clones and then throw all but one away in a manner which

emphasises certain events and de-emphasises others. There is a significant weakness in this reduction of importance sampling to importance splitting, however, in that the process of massive cloning followed by massive splitting is not likely to be an efficient way to distort the evolution of the process under study. Since the whole point of either of these methods is to gain accurate estimates more efficiently, it does not make sense to use an inefficient approach when a more efficient approach is available.

5 NUMERICAL EXAMPLES

5.1 A SIMPLE GAUSSIAN QUEUE

To start with, let us consider a simulation of a queue, the input to which is a series of Gaussian numbers with mean -2 and standard deviation 1. These have been simulated using quantum simulation in the form described in Subsection 2.2.2. The results are plotted, together with the expected results from theory, in Figure 1. The simulation results and the theoretical results overlap almost perfectly. This simulation included ten simultaneous threads, one of them completely conventional, and the others distorted in the manner of importance sampling, as described in Subsection 2.2.2.

A large deviations principal is known for this model [3], which leads to the conclusion that the optimal importance sampling simulation will make use of distorted probability measure, γ, in which the input to the queue becomes IID Gaussian with mean 2 and standard deviation 1.

The duration of this simulation was 100 cycles of input, buffering, and service, all of which is modelled by the equation:

$$B_{t+1} = (B_t + X_t)^+$$

in which $\{B_t\}$ denotes the contents of a buffer at time t, starting with $B_0 = 0$, X_t denotes the *net* input to this buffer, which is an independent and identically distributed sequence of Gaussian random variables with mean -2 and standard deviation 1, and $(\cdot)^+$ denotes $\max(0,\cdot)$.

Figure 2 shows the manner in which the weight of *one* of the distorted threads changes during the simulation. The weight of a thread reduces steadily due to the fact that the Radon-Nikodym derivative tends to decrease steadily. Soon after the weight of a thread becomes the lowest of all the weights of threads in the simulation, the thread is likely to be thinned, at which time it appears to have returned to a weight near one, in the plot shown in the figure. In fact what has happened is that this thread has been terminated, and the graph then shows the weight of a thread which has been produced by cloning the top thread. This thread therefore starts with weight close to 0.5 and reduces steadily, again, to a value near 10^{-40}.

The elapsed time for the simulation discussed here to complete, as implemented in mathematica, running under Linux, was less than one minute.

These simulations have also been carried out using quantum simulation in the form where cloning and thinning are used but no distortion at all. The results are similar, although not quite so accurate and the computation time to achieve these results was significantly longer. The transient behaviour of the same queue was also observed by means quantum simulation with a view to observing the speed with which the system approaches its stationary state. For more details in each case, see [2].

5.2 THE POISSON-PARETO BURST PROCESS

In this example a similar technique to that used in subsection 5.1 will be used in application to a much much more difficult problem: buffering of the Poisson-Pareto Burst Process. This process is formed as an a collection of bursts, the starting times of the bursts forming a Poisson process and the lengths of the bursts, each of which are statistically independent, Pareto distributed.

This process is particularly difficult to simulate because the natural evolution through a representative subset of the state space *of the traffic process itself* takes a very long time.

When a *bad* state occurs, because this will necessarily be caused by the simultaneous arrival of an unusually large number of *long* bursts, hence this state will tend to persist for quite a long time. To be a little more specific, suppose we are interested in a collection of states which occurs with frequency lower than 10^{-5} and that these states persist for at least 10000 cycles (as is the case for the Poisson Pareto Burst Process). It follows, for consistency with the expected frequency, that the average period of time *between* these events must be $10^5 \times 10000 =$ one billion cycles. In order to estimate probabilities lower than 10^{-5}, a conventional simulation would have to be several times longer than one billion cycles. How much longer is unclear, although it *is* clear that in order to estimate the probabilities of events which are less likely than this even longer simulations would be required.

In the *fast* simulation that we wish to undertake we will need to explore this state space by *starting* our simulations in a collection of states which explore the underlying state space of the traffic process *systematically*. Randomly exploring the state space might not be adequate. Furthermore, we can't afford to allow the natural dynamics of the process to take charge from that point on. We need to increase the frequency of certain paths of evolution – the ones where the buffer fills up.

The following techniques have been used to speed up the simulation and increase its accuracy:

(i) each initial thread of the simulation is allocated a certain number of *long bursts* (bursts whose remaining duration is longer than the entire simulation), and then assigned a weight in accordance with the probability of this event occuring; notice that in this respect

Figure 1: Simulation results and theoretical estimate for the complimentary waiting time distribution in a Gaussian queue (mean -2)

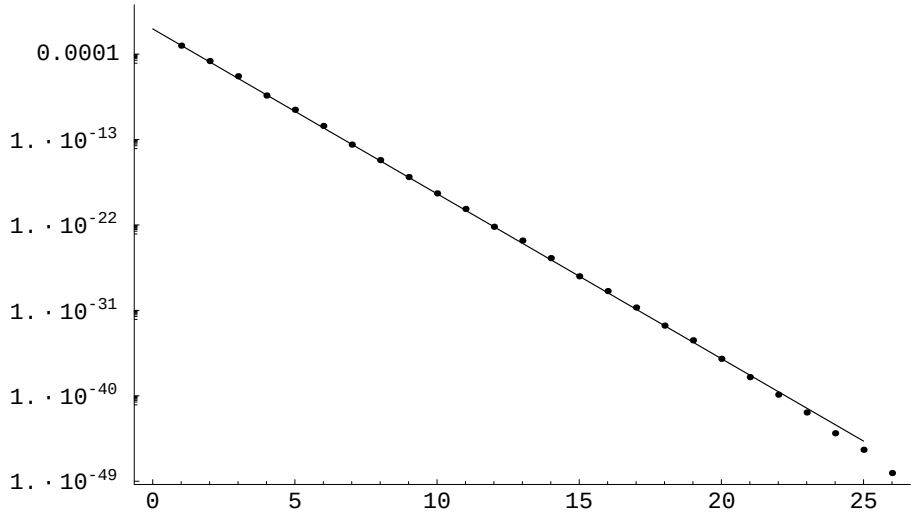
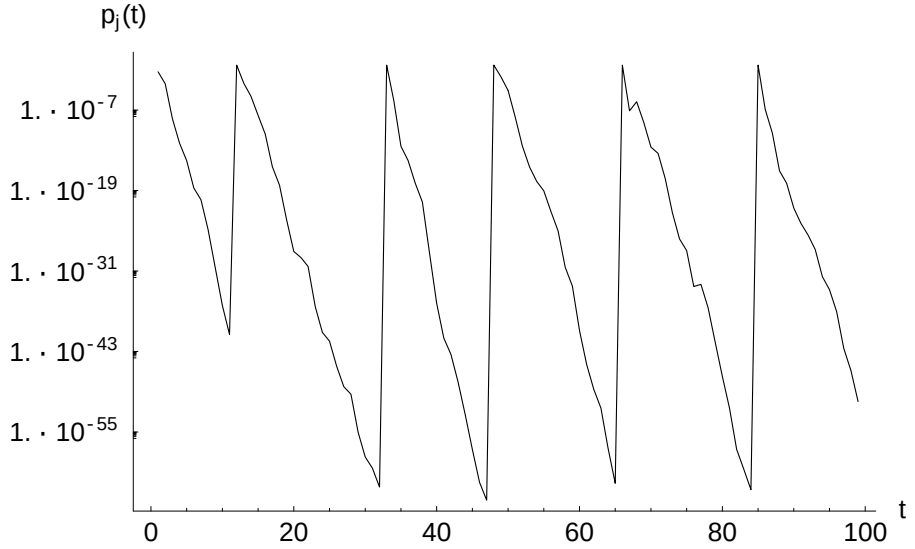


Figure 2: Variation in Weights during a Quantum Simulation of a Gaussian Queue



the state space is being explored systematically rather than randomly;

(ii) each initial thread is also allocated a number of existing bursts which are extant at the time when the simulation starts. These *short bursts* (bursts shorter than the entire simulation) are randomly selected according to a distorted distribution. Note that both the short and the long bursts which are extant at the start of the simulation are drawn from the forward recurrence time distribution corresponding to the Pareto distribution rather than from the Pareto distribution itself;

(iii) each thread *evolves* according to a *variety* of randomly selected distorted distributions;

(iv) on a regular basis, a collection of *candidates* are selected for thinning and from these candidates a subset (up to nine tenths) were selected for thinning. The thinned candidates are then replaced by *clones* which come from the remaining threads, i.e. the non-candidates. The choice of *candidates* is made so as to reduce the number of unpromising threads and increase the number of promising threads. Since the type of distortion present in threads varies randomly, this Darwinian selection helps to find the *right* types of distortion for the most efficient and accurate observation of the behaviour we are interested in.

(v) together with these threads which begin with a collection of existing long bursts and short bursts drawn from a distorted distribution and which evolve according to a distorted distribution there is a also one normal (undistorted) process which absorbs the weight which is lost from all the other threads. This thread may also generate clones which then evolve according to distorted rules of evolution;

(vi) all the above is repeated several times, completely independently, so that estimates of accuracy of the results obtained from the simulations can be obtained.

Some results of a quantum simulation using just these techniques are depicted in Figures 3 and 4. These plots show results from a quantum simulation of a buffer fed by a Poisson Pareto burst process with $\lambda = 100$ (the intensity of the Poisson process), $\delta = 1$ (the minimum burst length), $r = 0.02$ (the intensity of a burst), $\gamma = 1.5$ (the exponent parameter of the Pareto distribution of the bursts), and a server sufficiently fast that the the net mean input in one unit of time is -1.

The plots shown in Figure 3 shows an estimate of the buffer exceedence probabilities obtained after a simulation of length 50 (i.e. 50 time steps), together with confidence intervals obtained by conducting 10 completely independent simulations and choosing the largest estimate as the upper level and the lowest estimate of the ten simulations for the lower level of a confidence interval. Each of the ten independent simulations contained 100 simultaneous threads at any one time, which were thinned and cloned from time to time during the simulation. Figure 4 depicts a sequence of snapshots of the estimated buffer exceedence distribution at times separated by 10 units of time, starting from an empty buffer and finishing after 50 units of time.

Thus, the width of the confidence intervals in Figure 3 gives us confidence that the results obtained from the quantum simulation are consistent and are achieving a satisfactory accuracy, while the convergence of the results which is apparant in Figure 4 gives us confidence that the simulations are sufficiently long and have converged to a stable result by the end of the simulation.

These experiments were also repeated several times and produced similar results on each occasion.

There are also two theoretical curves shown in these plots. The lower curve is a Gaussian approximation for this queueing system, obtained using the results of [3]. It is expected that the buffer distribution in a system with Poisson Pareto burst input should approach the Gaussian system as the Poisson rate increases, with fixed rate, r. However, the rate of convergence is unknown. The upper curve is a theoretical approximation for the buffer distribution in this sytem, obtained by means of a quasi-stationary approximation in which the slowly varying state of the system models the current number of long bursts (as long as the entire simulation) and the rapidly varying behaviour corresponds to a system in which bursts are limited in length (shorter than the entire simulation), which can be modelled with reasonable accuracy as a Gaussian system, using the results from [3]. This theoretical approximation for queueing behaviour of a system with Poisson Pareto Burst input has been explored in more detail in [4].

The key difficulty in simulating this process is that the initial state can have a very significant effect on the results derived from the simulation, and addressing this difficulty simply by increasing the number of threads is not sufficiently effective. Instead of simply increasing the size of the initial population, a better approach is to systematically explore the initial states. This is precisely what has been done in the case of the number of *long bursts* – the possibility that 0, 1, 2, etc long bursts occur is explored, up to a sufficiently large number.

Conventional simulation of this system can also be effective if the same approach to long and short bursts is used, although a conventional simulation will probably not be able to accurately estimate probabilities below a certain level, eg about 10^{-6}. Conventional simulations have been used for the system which has been investigated in this paper and the conventional and quantum simulations have been found to be in good agreement.

Figure 3: A Quantum Simulation of the Poisson Pareto Burst Process – simulation results with confidence intervals

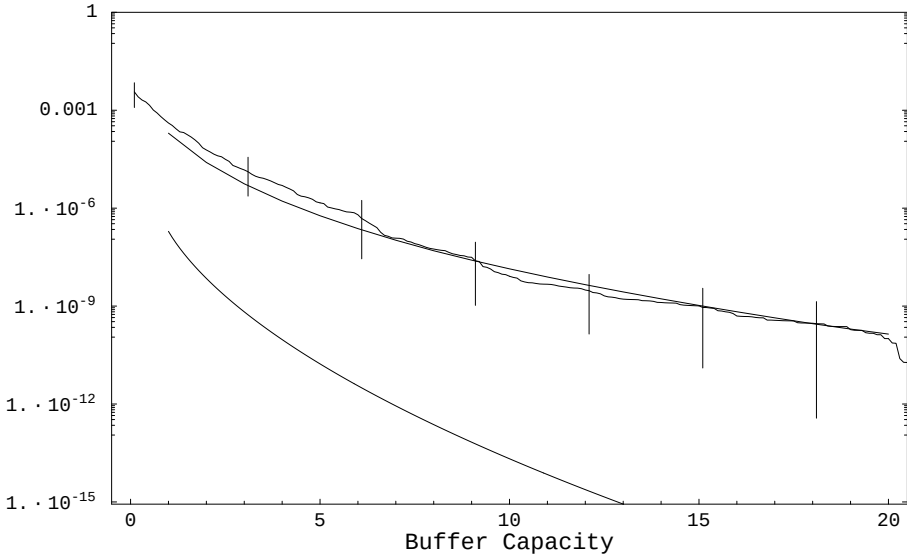
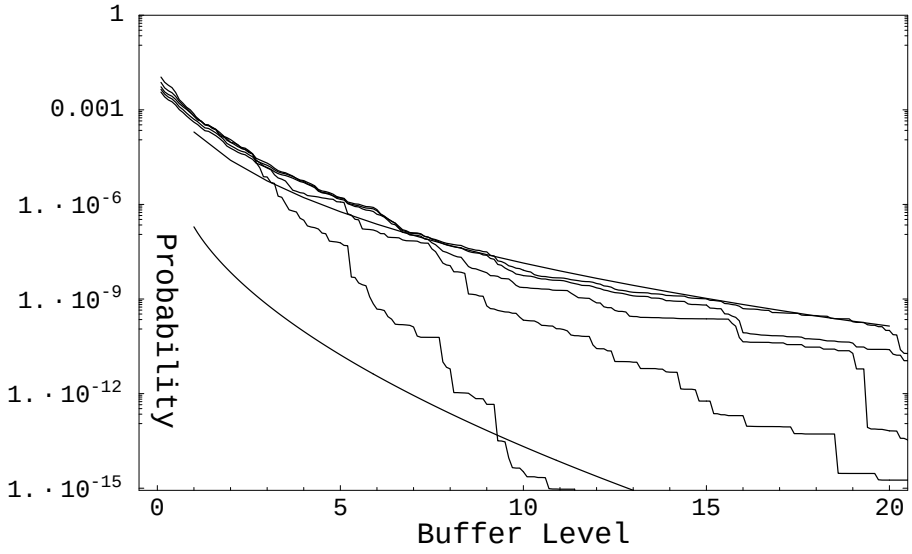


Figure 4: A Quantum Simulation of a PPBP queue – results at times 10, 20, 30, 40 and 50 and comparison with two theoretical results: the quasi-stationary approximation and a Gaussian approximation



6 CONCLUSION

A technique of rare event simulation referred to here as quantum simulation has been described, compared with the existing rare event simulation methods of importance sampling and importance splitting, and with other closely related techniques. Quantum simulation with precisely one thread is equivalent to importance sampling and quantum simulation in which cloning occurs at certain regions of the state space is equivalent to importance splitting. The full range of techniques which can be legitimately used in a quantum simulation to produce more accurate estimates is difficult to specify exhaustively, however, a natural requirement which all simulations should satisfy is that unbiased estimates can be made of any well defined statistical parameter or description applicable to the original simulation. An equation which states this basic requirement was defined and used in the rest of the paper to discuss and validate a wide range of possible techniques which can be used. This equation is the defining characteristic of technique used in this paper and has been used to guide the techniques used in the numerical examples.

Rules which ensure that cloning and thinning procedures retain the consistency property are then defined. It is important that these rules be stated in as general a form as possible. In particular, it is shown that thinning can be done by weighted sampling, either for the threads to thin, or for the threads to retain, although the weights are different in each case. However, useful as either of these strategies are, this is not the only possible valid and useful scheme for thinning which is consistent with the basic consistent estimation rule.

The importance of the thinning rules is that they can be used to preferentially select *better performing* threads. What it means for a particular thread to be performing better than another will vary considerably from case to case.

Two numerical examples have been used to demonstrate the effectiveness of quantum simulation and to demonstrate the relationship between the importance sampling style of quantum simulation and the importance splitting style. The second example, which was the more difficult, was of a system which is very difficult to simulate by conventional means – a server and buffer fed by a Poisson-Pareto burst process. The accuracy of the quantum simulations was supported in this case by comparison with a theoretical model which has independently been validated against conventional simulation. Although the quantum simulation run lengths were extremely short in simulated time they produced accurate estimates of queue length exceedence probabilities down to 10^{-10}. The conventional simulations [4] by comparison produced accurate estimates down to the level of probabilities as low as 10^{-6}.

The three techniques: conventional simulation, theoretical approximation, and quantum simulation, have each provided a significant contribution to a deeper understanding of this important problem. The theoretical approximation, was, for example, suggested by the approach used in the quantum simulation. The agreement between all three techniques provides strong evidence that all three techniques are valid models of the real system in their particular domain of accuracy

How can a quantum simulation consistently mimic every aspect of the real system, i.e. produce unbiased estimates of every aspect of the original system (time-varying, correlations, variances, etc.) while achieving *better* accuracy? If a variance were to be estimated from a quantum simulation, the mean value of this variance should be, surely, the expected value of this variance in the original system.

The answer is that the accuracy of estimates in any simulation, quantum simulations included, is a function not of the variance of the *original* system, but rather of the variance of the *estimates obtained* from the system. In a conventional simulation, estimates *are* often obtained as simple observations, or averages of observations over time. However, estimates obtained from quantum simulations are not observations in the usual sense: they are obtained as (weighted) averages already, even the simplest possible estimate, and the variance of one of these weighted averages is something which doesn't correspond to any obervation made directly on the original system.

Estimation of the variance of estimates in a quantum simulation can be obtained quite readily by simply conducting several completely independent quantum simulations (either simultaneously, or one after the other) and then using the conventional method for computing a sample variance. This is basically the method which has been used to compute confidence intervals for the simulations given in the numerical examples presented above.

Finally, let us consider the issue of optimal configuration of simulations. In the case of importance sampling, for example, the selection of the best distorted probability measure is quite important, and similar issues occur in the context of importance splitting. In the case of quantum simulation, the simulation itself searches for the best parameters by a genetic algorithm which is applied in the thinning stage. However, it needs to be kept in mind that in many simulations there is not, in fact, a single over-riding objective. For example, in the case where we wish to estimate a probability distribution, we will usually want to estimate the entire probability distribution, not just one value. The "optimal" choice of parameters for the quantum simulation may need to be a compromise taking into account the diverse matters of interest.

REFERENCES

[1] R. G. Addie. Quantum simulation - rare event simulation by means of cloning and thinning. In *Proceedings of MODSIM 2001*, December 2001.

[2] R. G. Addie. Quantum simulation - rare event simulation by

means of cloning, thinning and distortion. Technical report, University of Southern Queensland, Department of Mathematics and Computing, 2002.

[3] R. G. Addie, P. Mannersalo, and I. Norros. Most probable paths and performance formulae for buffers with gaussian input traffic. *European Transactions on Telecommunications*. Accepted for publication.

[4] R. G. Addie, T. M. Neame, and M. Zukerman. Performance evaluation of a queue fed by a Poisson Pareto burst process. *Communication Networks*, 2002. Accepted for publication.

[5] Ahmet A. Akyamac, Zsolt Haraszti, and J. Keith Townsend. Efficient rare event simulation using DPR for multidimensional parameter spaces. In P. Key and D. Smith, editors, *Teletraffic Engineering in a Competitive World*, volume 3B of *Teletraffic Science and Engineering*. 16th International Teletraffic Congress, Elsevier, 1999.

[6] David M. Ceperley. Lectures on Quantum Monte Carlo. Technical report, National Centre for Supercomputing Applications (NCSA), 1996.

[7] Dan Crisan and Arnaud Doucet. Convergence of sequential Monte Carlo methods. Technical report, University of Cambridge, Cambridge, 2001.

[8] Michael R. Frater, Robert R. Bitmead, Rodney A. Kennedy, and Brian D. O. Anderson. Fast simulation of rare events using reverse-time models. *Computer Networks and ISDN Systems*, pages 315–321, 1990.

[9] Bárbara González-Arévalo and Gennady Samoridnitsky. Buffer content of a leaky bucket system with long-range dependent input traffic. Technical report, Cornell University, 2000.

[10] Carmelita Görg and Oliver Füss. Simulating rare event details of ATM delay time distributions with RESTART/LRE. In P. Key and D. Smith, editors, *Teletraffic Engineering in a Competitive World*, volume 3B of *Teletraffic Science and Engineering*. 16th International Teletraffic Congress, Elsevier, 1999.

[11] P. Grassberger and W. Nadler. "Go with the winners"-simulations. Technical report, Heraeus Summer School, Chemnitz, 2000. http://xxx.lanl.gov/abs/cond-mat/0010265.

[12] Zsolt Haraszti and J. Keith Townsend. The theory of direct probability redistribution and its application to rare event simulation. In *Proceedings of IEEE Infocom*, 1998.

[13] Yukito Iba. Population-based Monte Carlo algorithms. Technical report, Institute of Statistical Mathematics, Tokyo, 2000.

[14] P. E. Lassila and J. T. Virtamo. Efficient importance sampling for Monte Carlo simulation of loss systems. In P. Key and D. Smith, editors, *Teletraffic Engineering in a Competitive World*, volume 3B of *Teletraffic Science and Engineering*. 16th International Teletraffic Congress, Elsevier, 1999.

[15] N. Likhanov, B. Tsybakov, and N. D. Georganas. Analysis of an atm buffer with self-similar ("fractal") input traffic. In *Proceedings, IEEE Infocom 1995*, pages 1–15. IEEE, April 1995.

[16] Minothi Parulekar and Armand M. Makowski. Tail probabilities for a multiplexer with self-similar traffic. In *Proceedings of Infocom '96*, pages 1452–1459, 1996.

[17] Boris Tsybakov and Nicolas D. Georganas. Overflow and losses in a network queue with a self-similar input. *Queueing Systems*, 2000.

[18] M. Villén-Altamirano and J. Villén-Altamirano. Restart: a method for accelarating rare event simulations. In J. W. Cohen and C. D. Pack, editors, *13th International Teletraffic Congress*. International Teletraffic Congress, North-Holland, 1991.